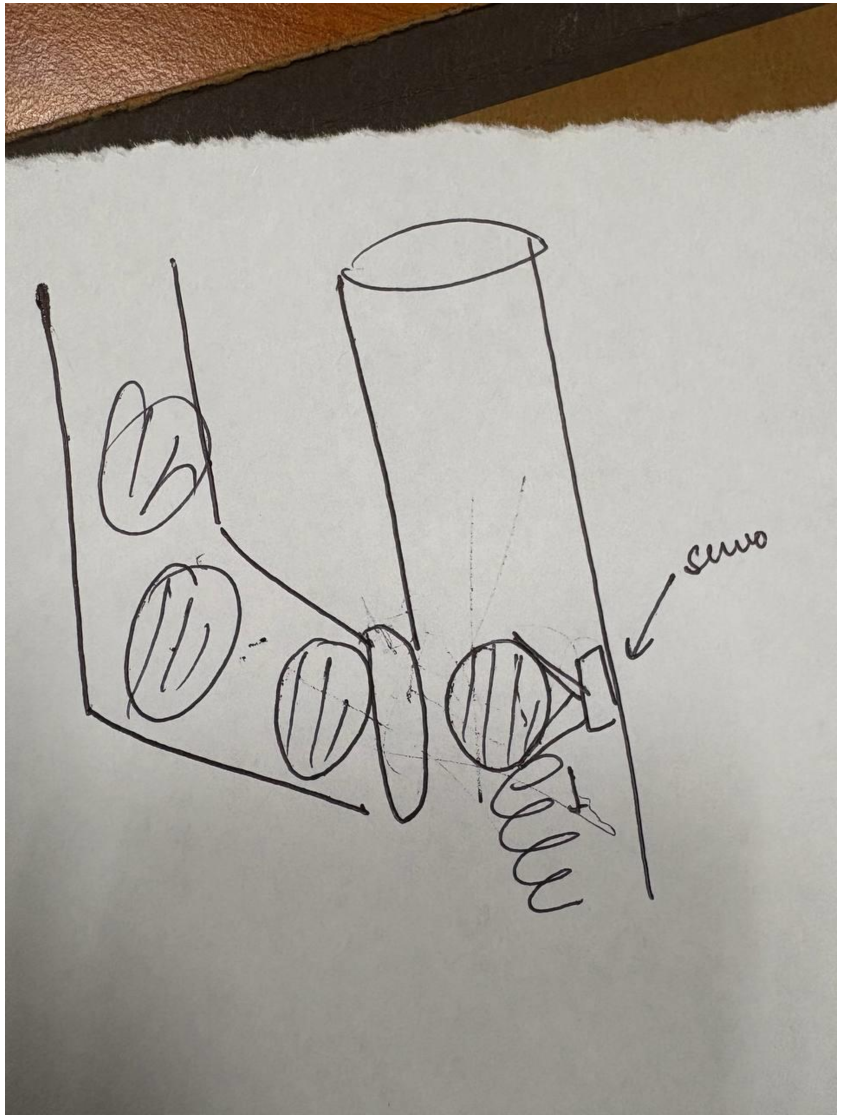
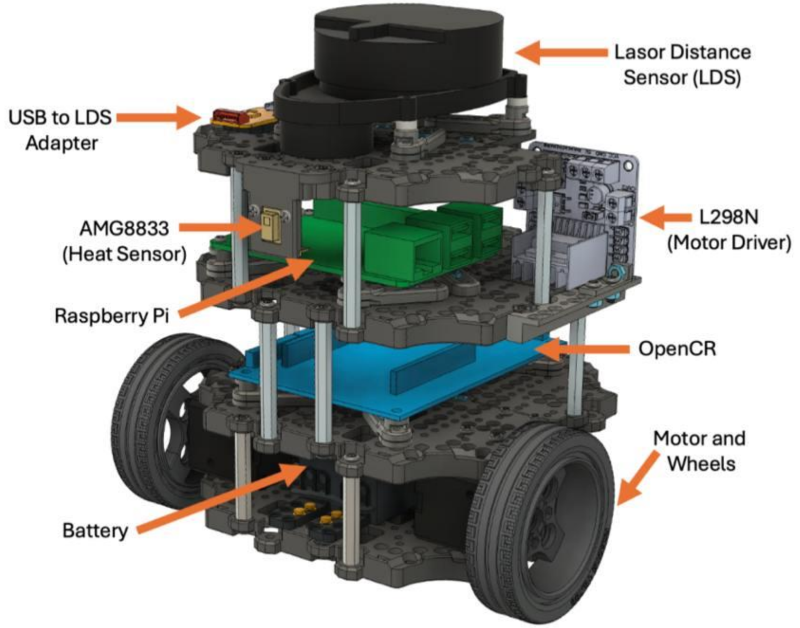
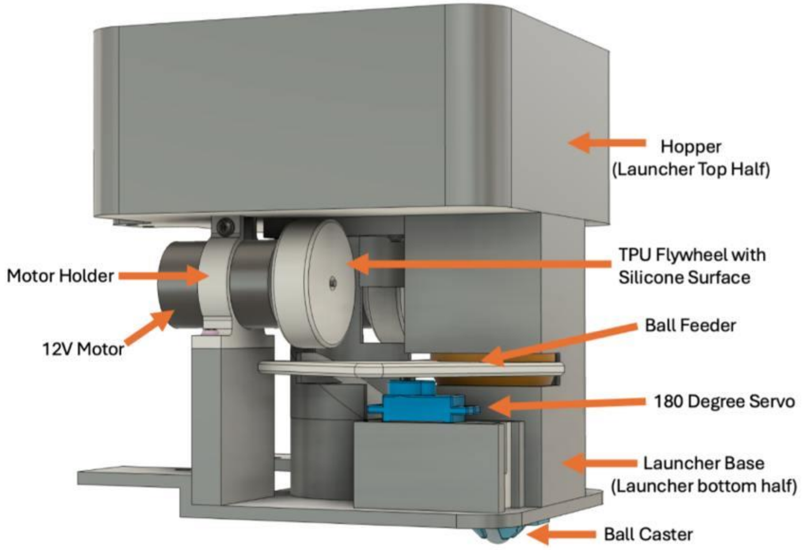
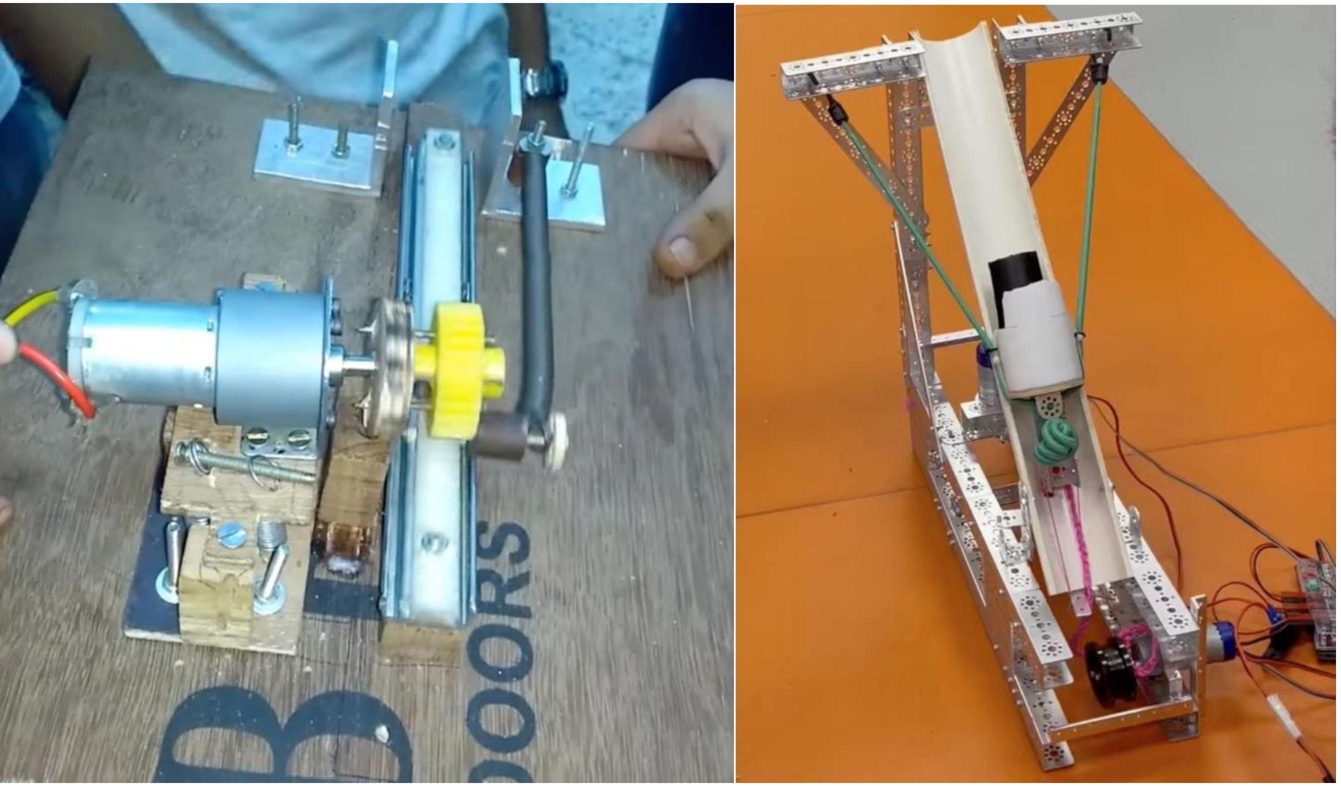
How we went from our first idea to our final design





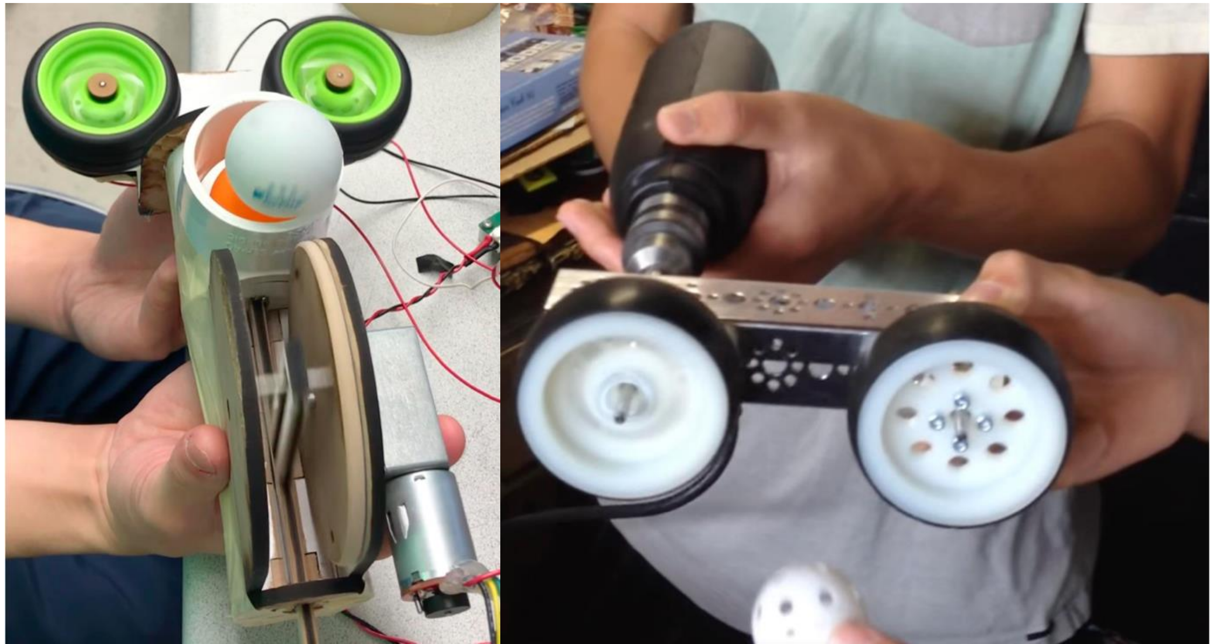


We researched various launching methods and unanimously agreed on a flywheel system



Motor that primes
spring or elastic band

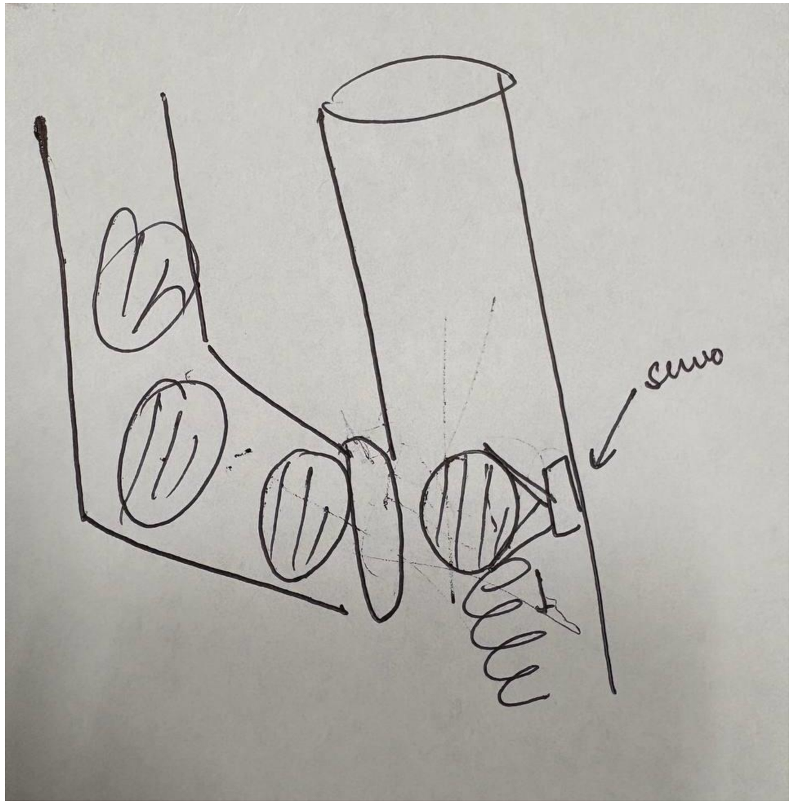
Motor that primes
spring or elastic band



Existing solutions did not meet our requirements

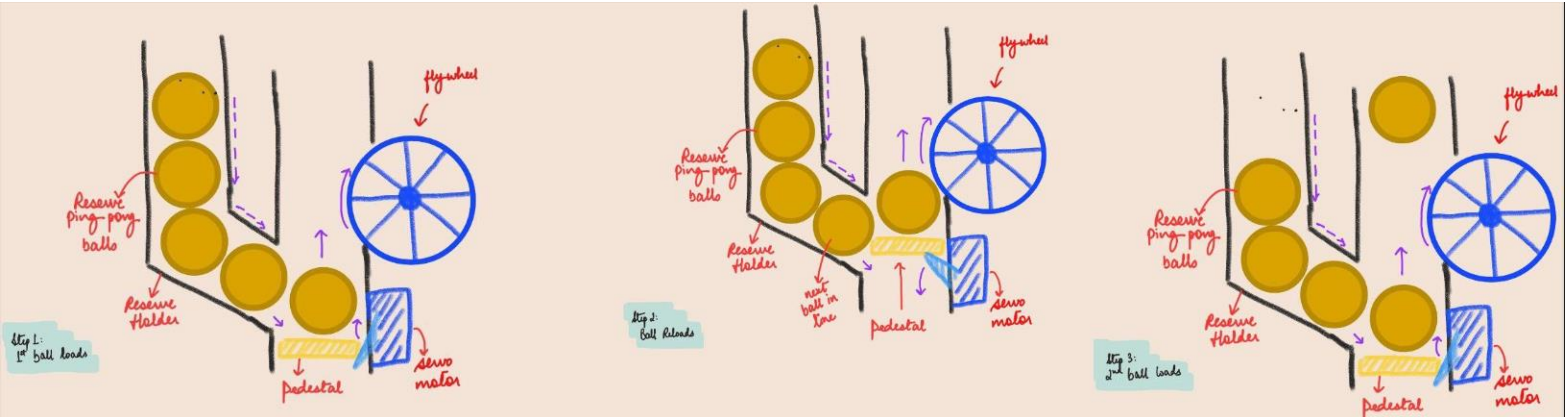
Existing solutions
- Most existing flywheel designs we found shot the ball forwards
- A gravity fed system for automatic loading worked

Our requirement
- Our ball needed to be launched vertically up
- We needed a way to feed the ball into the flywheels



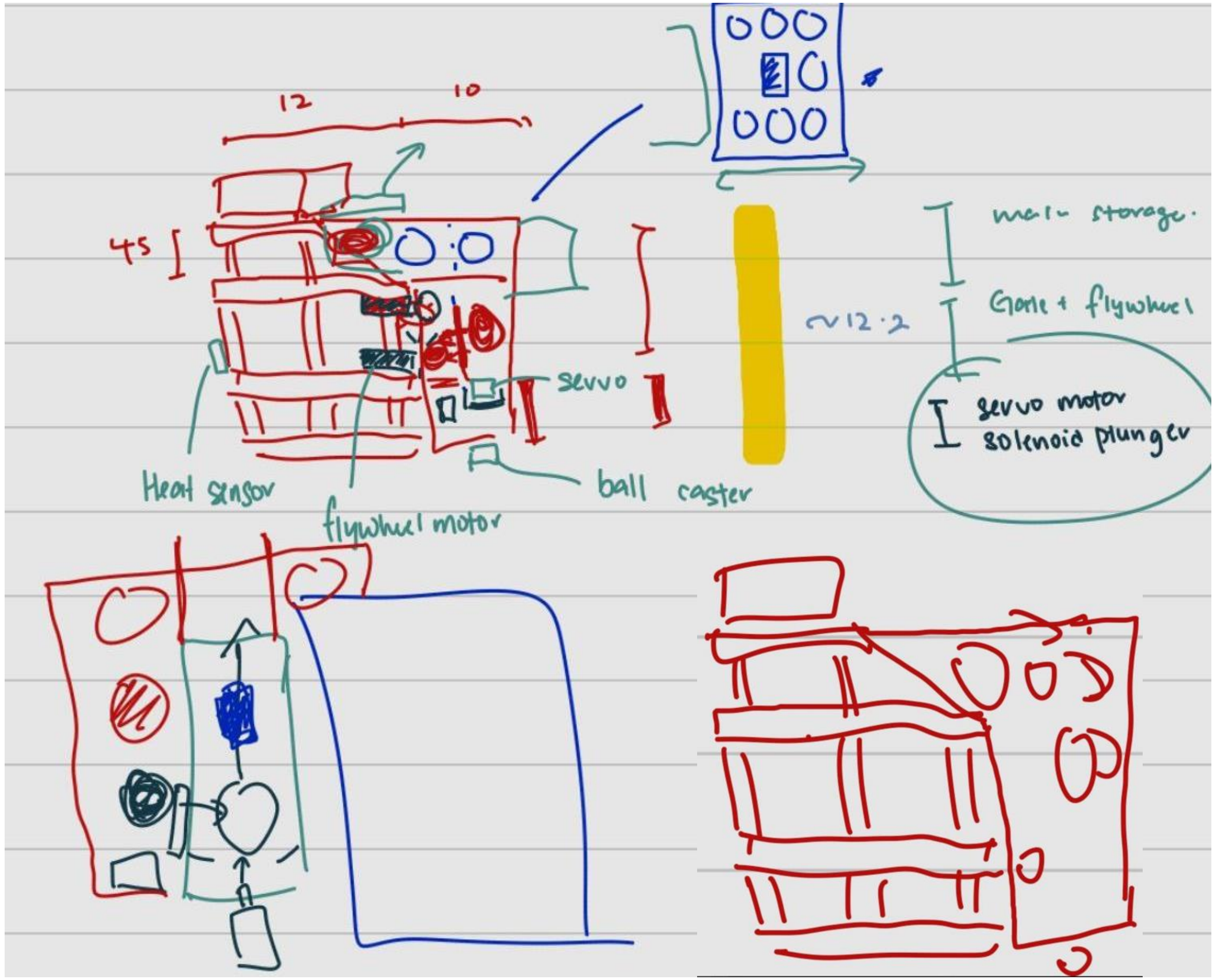
Sketch of our first idea to use a servo to push the ball up into flywheel

We developed our first idea



The payload is loaded onto a pedestal before a servo lifts it up to the fly wheel, launching it. The pedestal lowers for the next payload, repeating the cycle.

We thought about how such a mechanism could be mounted on the turtlebot



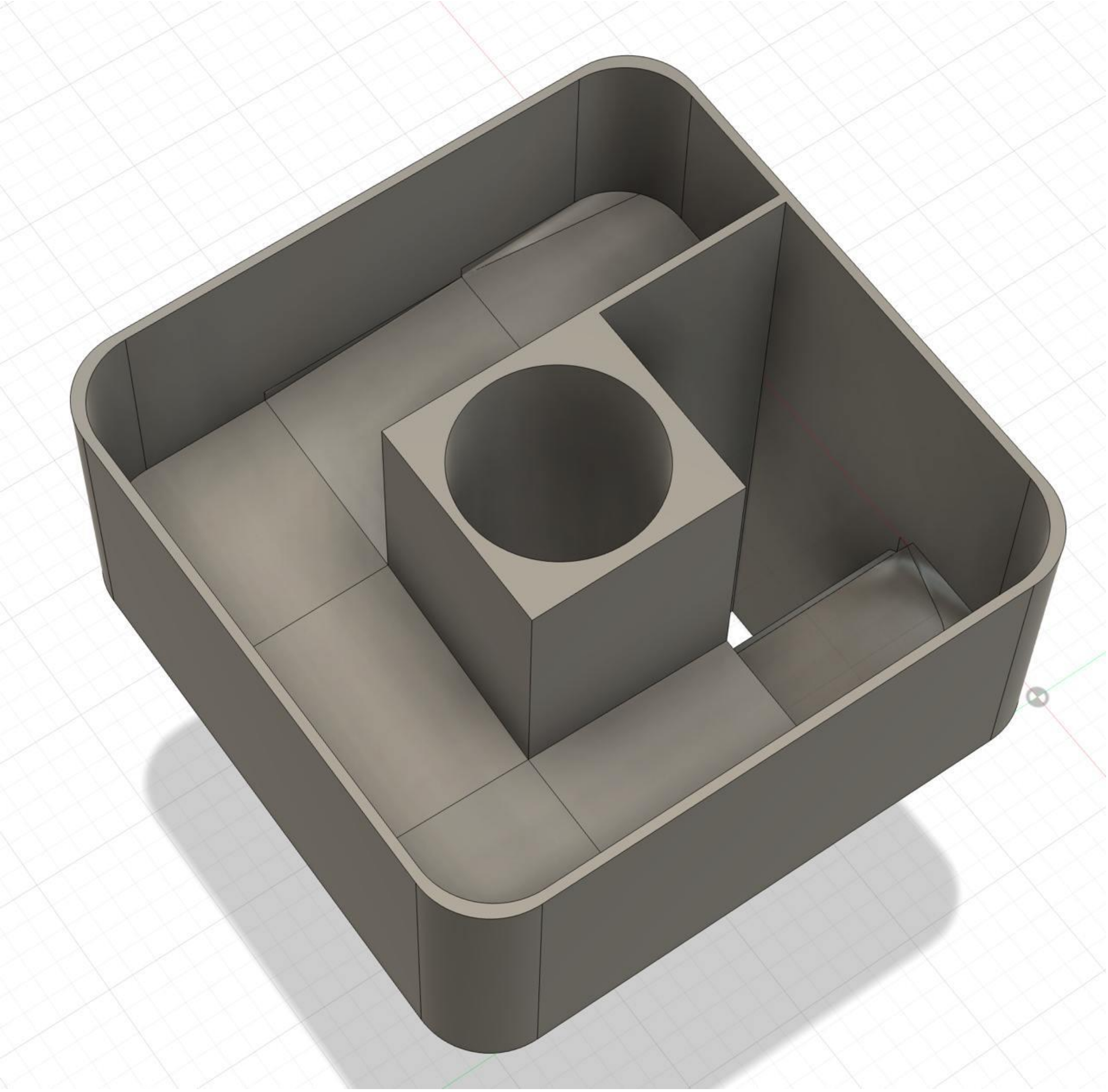
Storage for 9 balls:

- we thought of the 3x3 hopper system used in our final design that had the centre as the barrel

Mounting:

- We thought part of the hopper could mount on the turtle bot. To not block the LDS, we had to make space on Layer 3.

We modelled our first iteration of the hopper



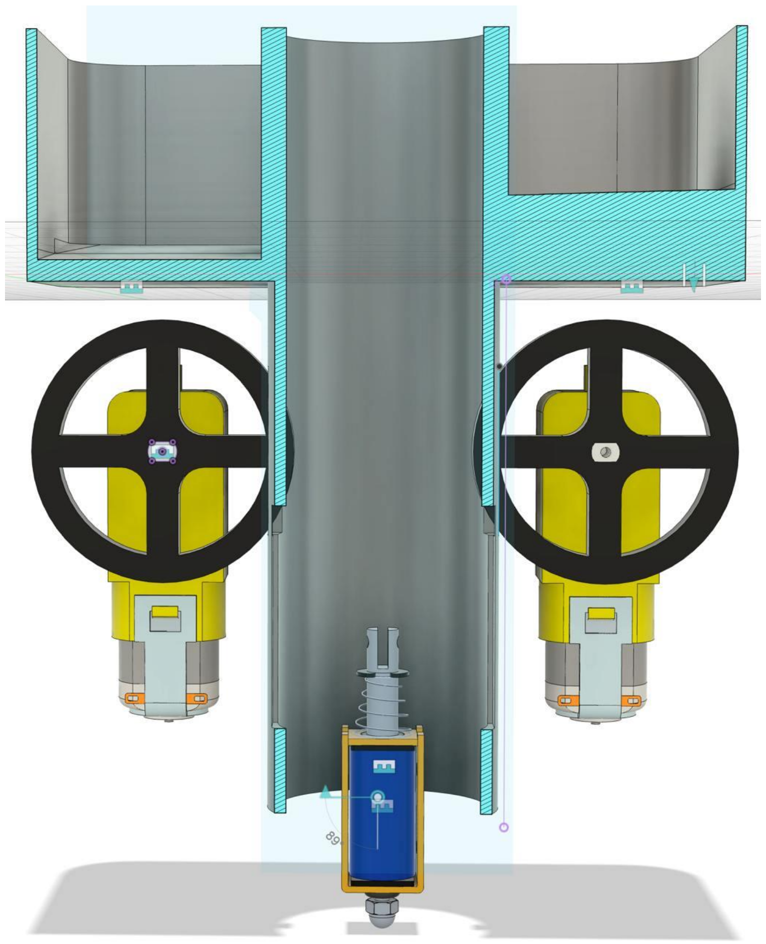
Jamming Consideration:

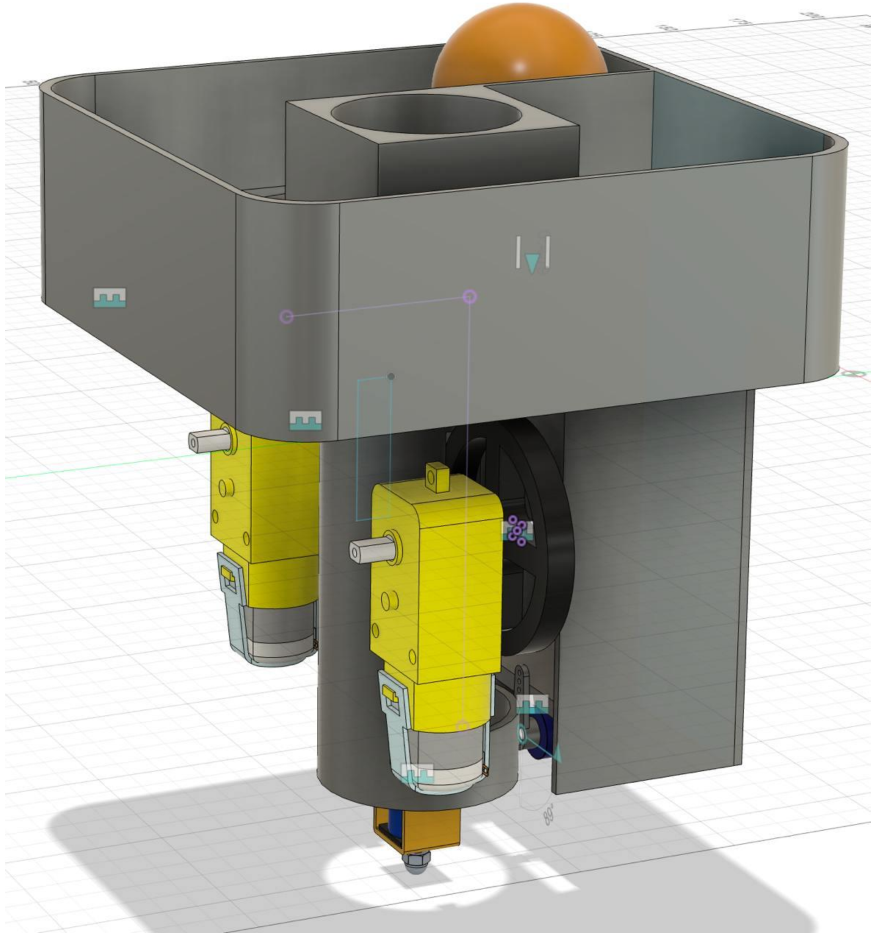
- Our design ensured balls only had one path to follow as we knew a hopper system would be susceptible to balls jamming when 2 or more try to enter the same opening at once.

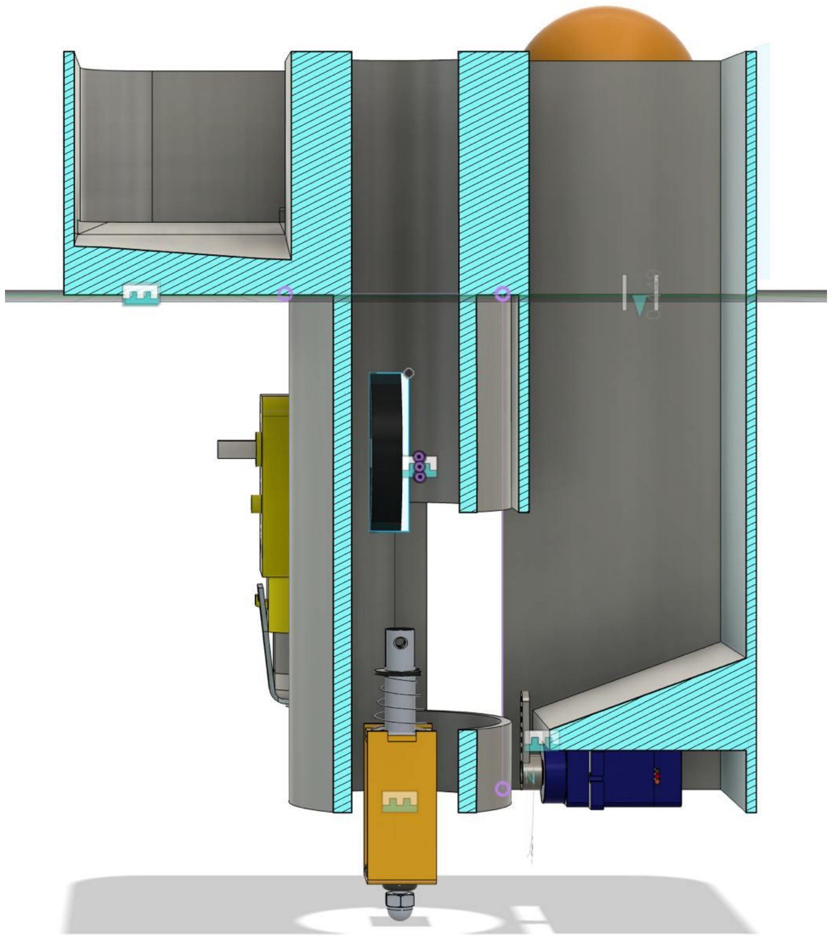
Slope:

- Unsure of the ideal slope height, we first incremented a 3mm height for each adjacent square

We designed the rest of the launching mechanism but still did not know how to mount/secure any components

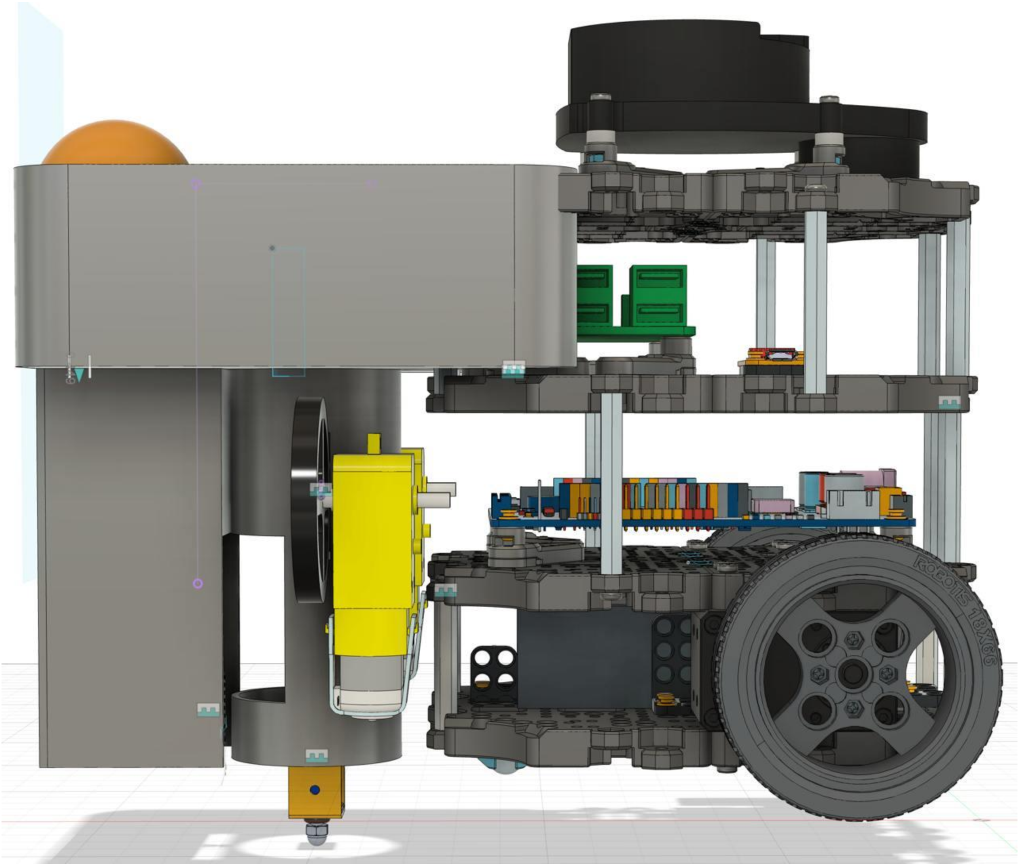




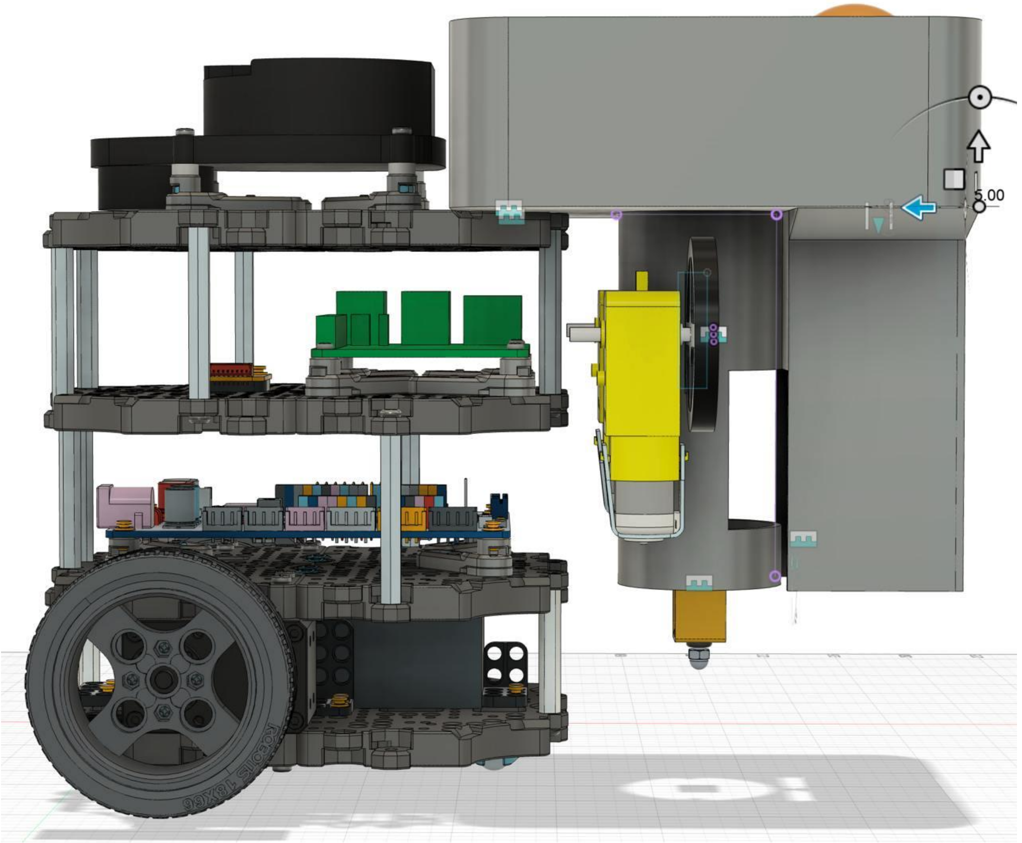


Parts were still floating in this iteration

We then realised this system utilising a solenoid was too tall
Mounting on neither Layer 3 nor 4 was feasible

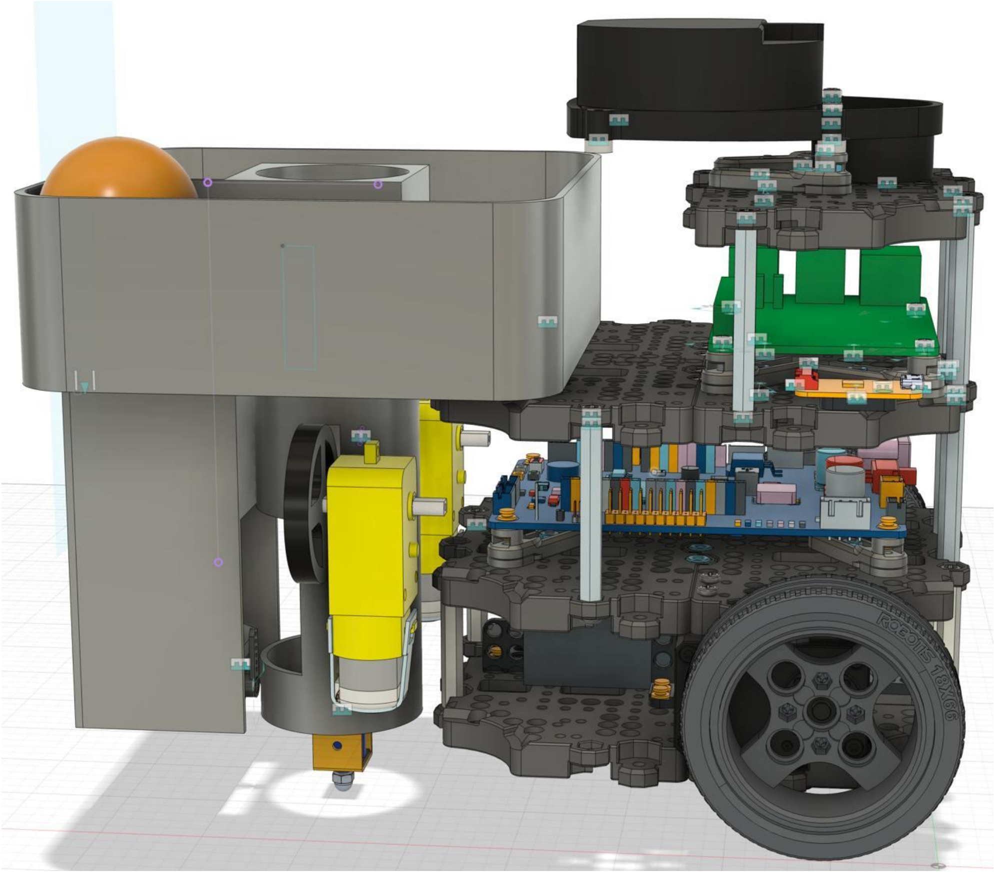


Mounting it on layer 3 meant exceeding the base of the turtlebot

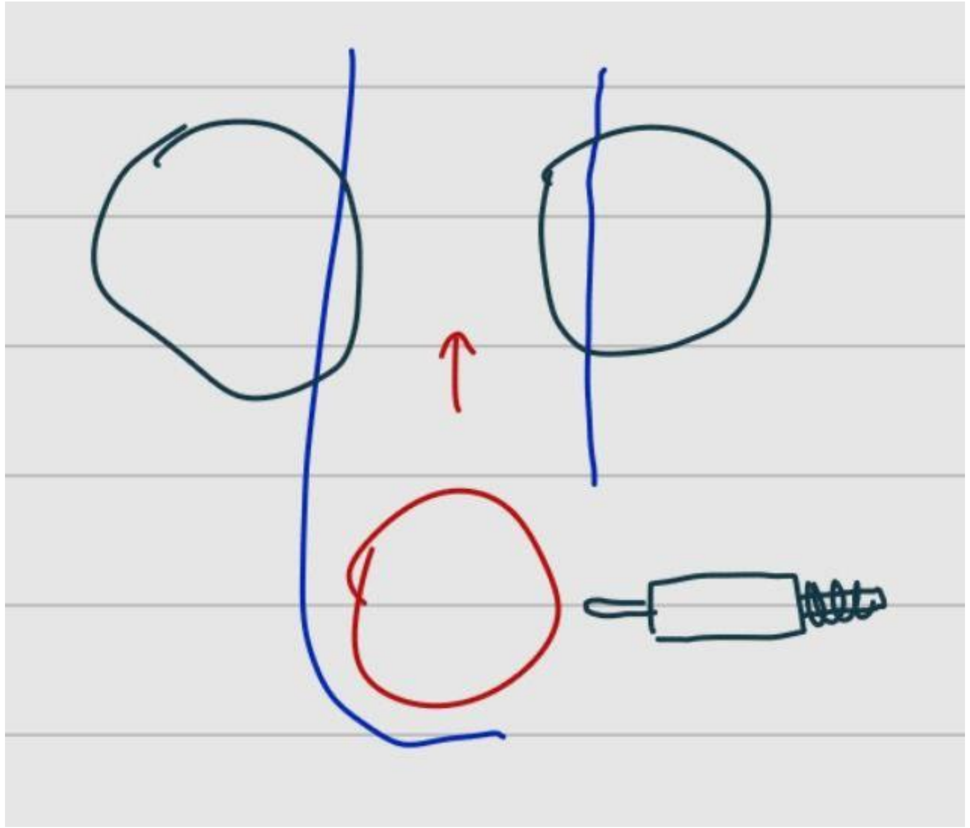


Mounting it on layer 4 meant blocking the back of the LDS

Deciding that blocking the LDS was not an option and that we would eventually figure out the height issue, we mounted on Layer 3

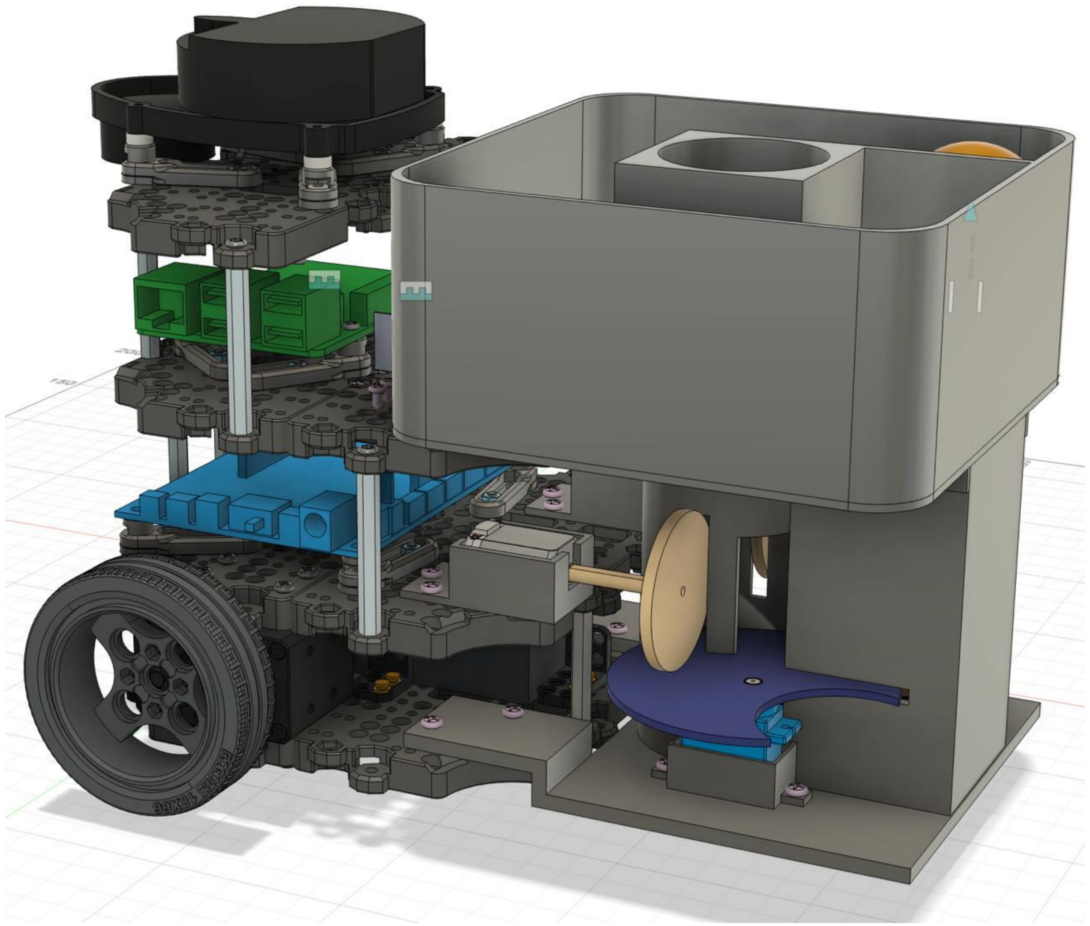


We shifted the RPI forward and removed a waffle plate to make space. But now the LDS was suspended

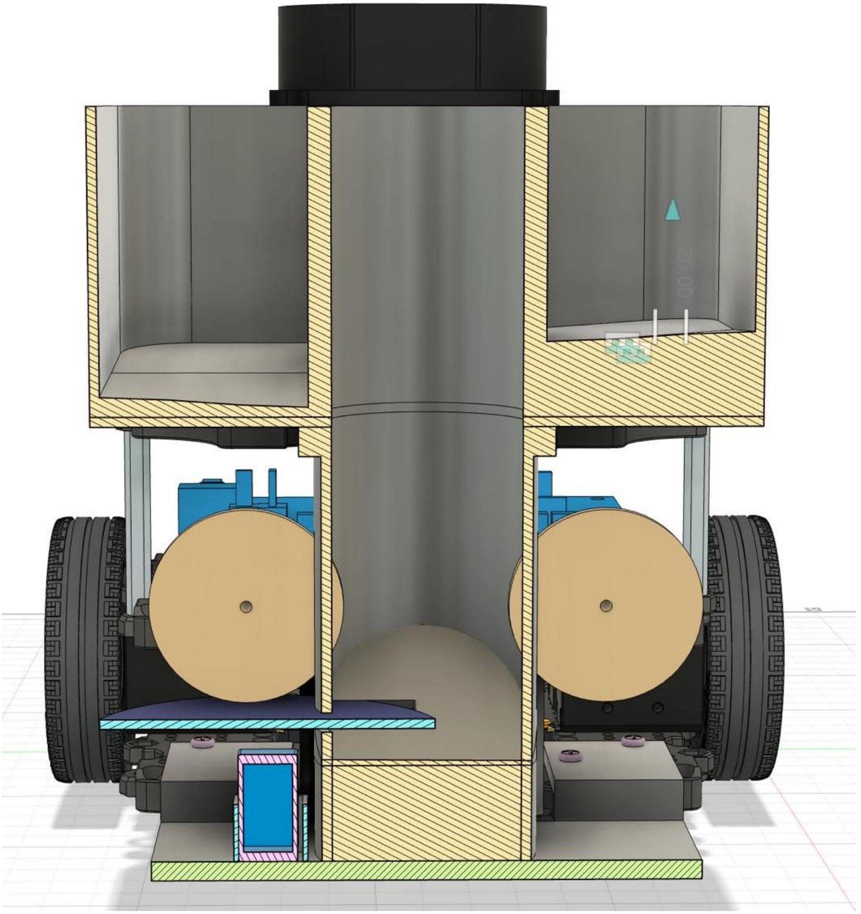


Instead of pushing the ball up directly, maybe we could push it horizontally onto an inclined surface to reduce height

We naively shifted the LDS forward to secure it and rotated the OpenCR to make space for motors. We updated the ball feeding mechanism.

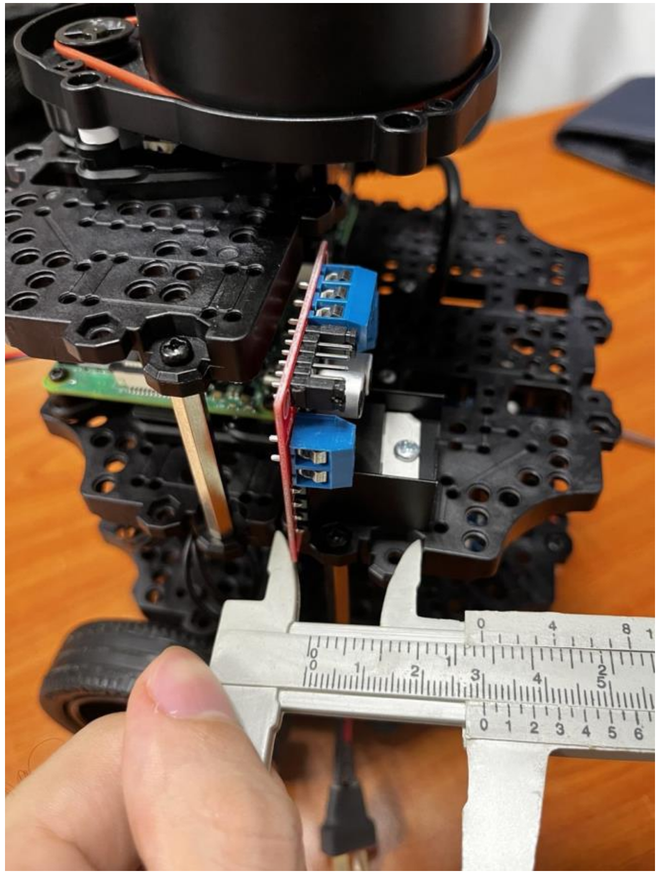


This looked great until we realised moving the LDS and OpenCR meant disastrous software implications



Feeding mechanism showed promises as it met our height requirements and worked in theory

The LDS needed to be supported but also could not be move. Removing a whole waffle plate was not an option, but what about half of one?

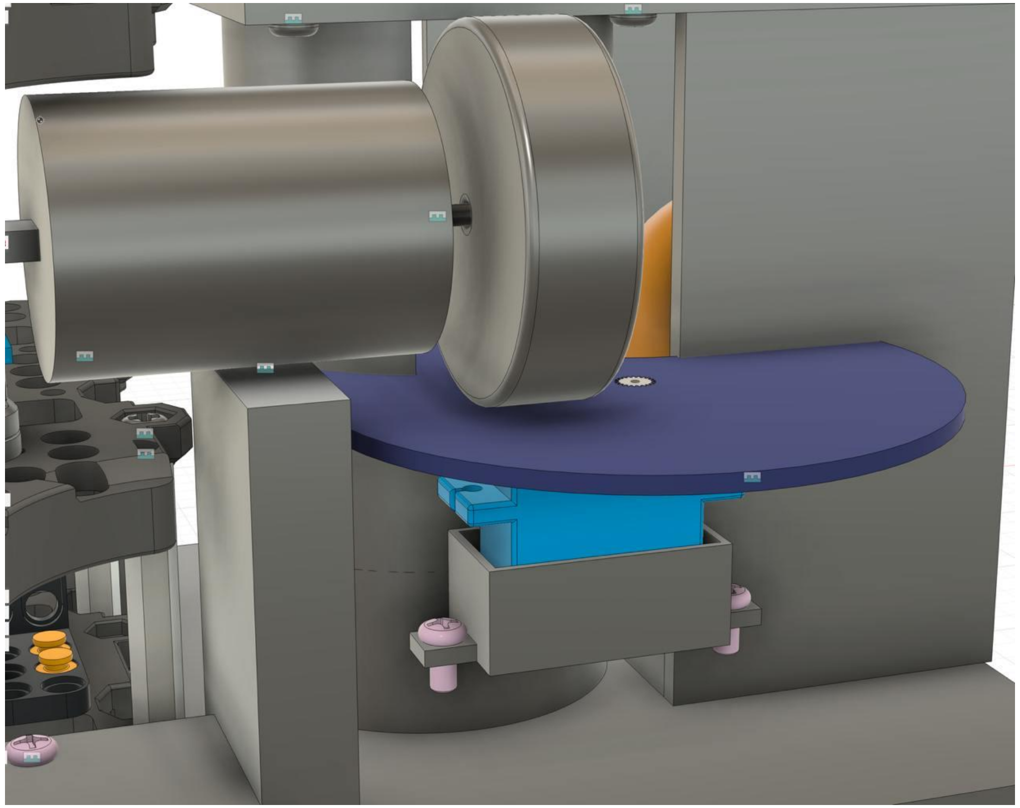


This space was needed for the L298N. If only we had a waffle plate that existed beneath the LDS only

*We eventually secured the L298N in a similar way

We 3D printed our own waffle plate that freed up space for the hopper and L298N while supporting the LDS

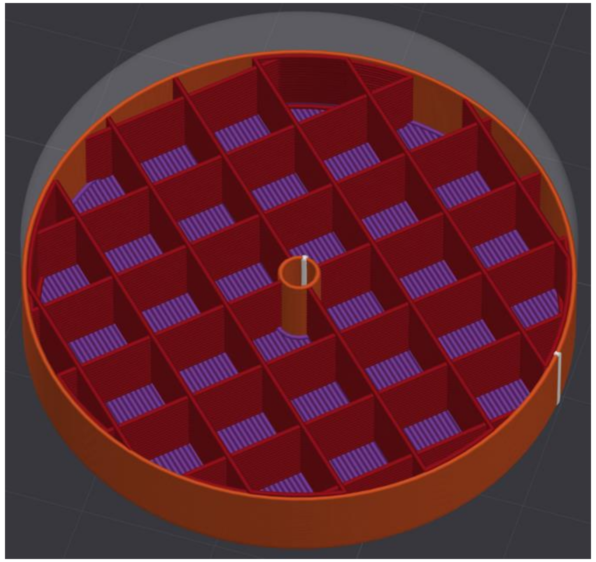
Without rotating the OpenCR, we had to find a space for our Motors



We raised a platform from the base of the launching mechanism. This brought the motor closer to where we needed the wheels to be, eliminating the need to extend the motor shaft

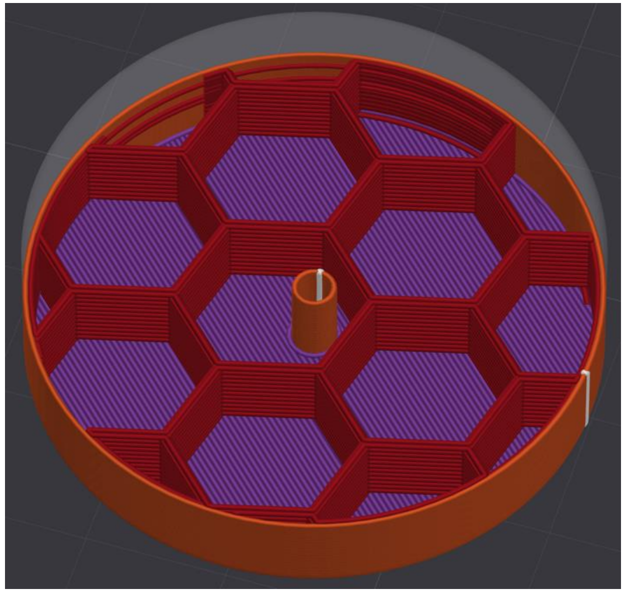
Taking inspiration from a motor holder design online. We custom made our own in TPU.

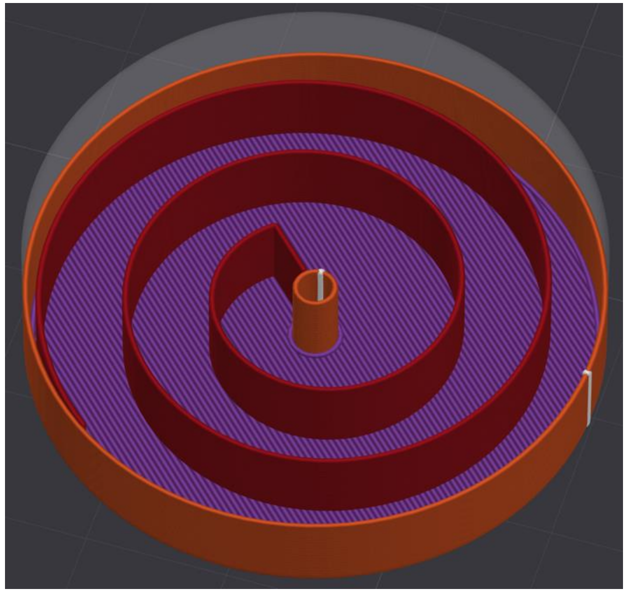
For flywheels, we needed good grip and some squeeze when the ball passed between the flywheels. We tested various 3D print settings



Rectilinear infill pattern, no “squeeze”

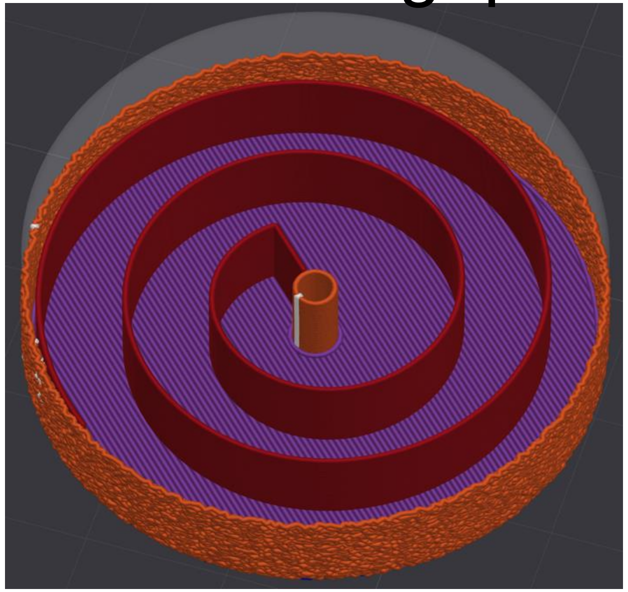
Honeycomb infill pattern, no “squeeze”





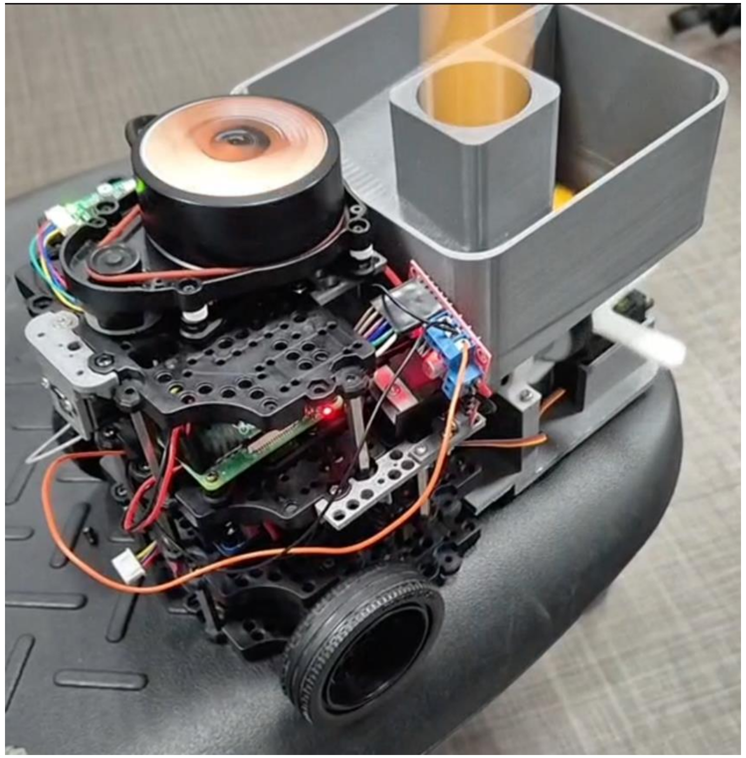
Archimedean Chords infill Pattern, good “squeeze”

Fuzzy Skin added to create rough surface for better grip



Lastly, we added black gasket to the surface to further enhance the grip

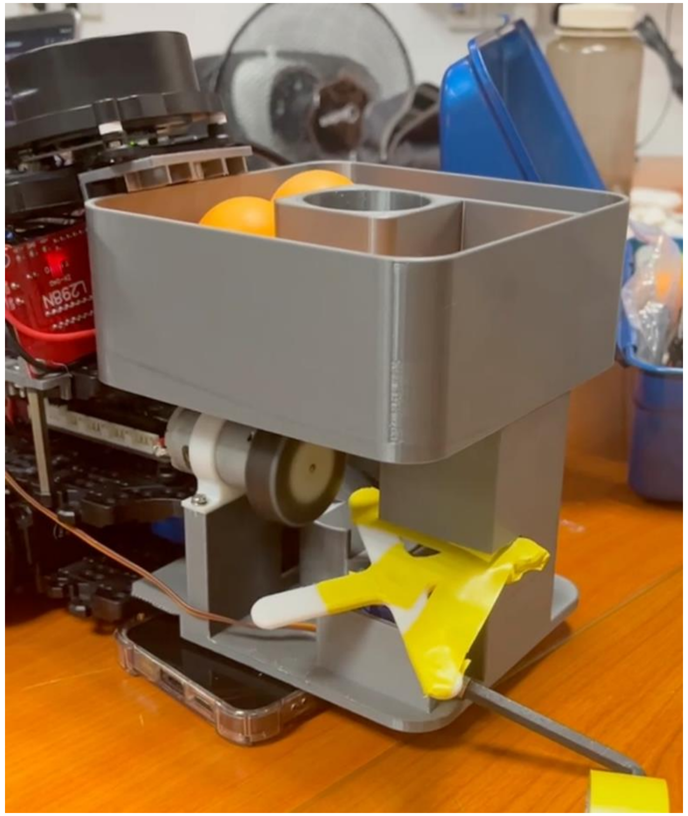
At this point we had fabricated the parts and assembled the bot including electrical components. One big issue remained.



The flywheels worked well and the hopper system worked as balls could be launched well consecutively.

The mission required the ball to be launched at precise intervals. But the 360 servo we used did not have precise positional control, meaning we were unable to reliably move it at the angles we needed. This meant we had no control over how many balls we could shoot.

We needed a positional servo for precise control but we could not acquire one that was also 360. We improvised.

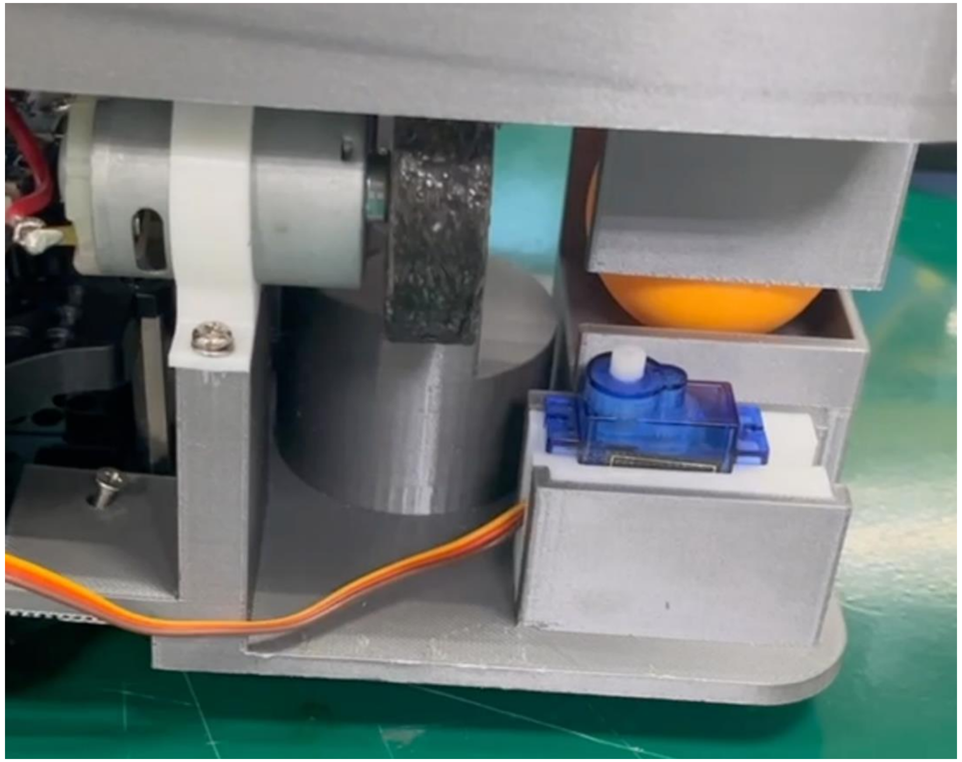


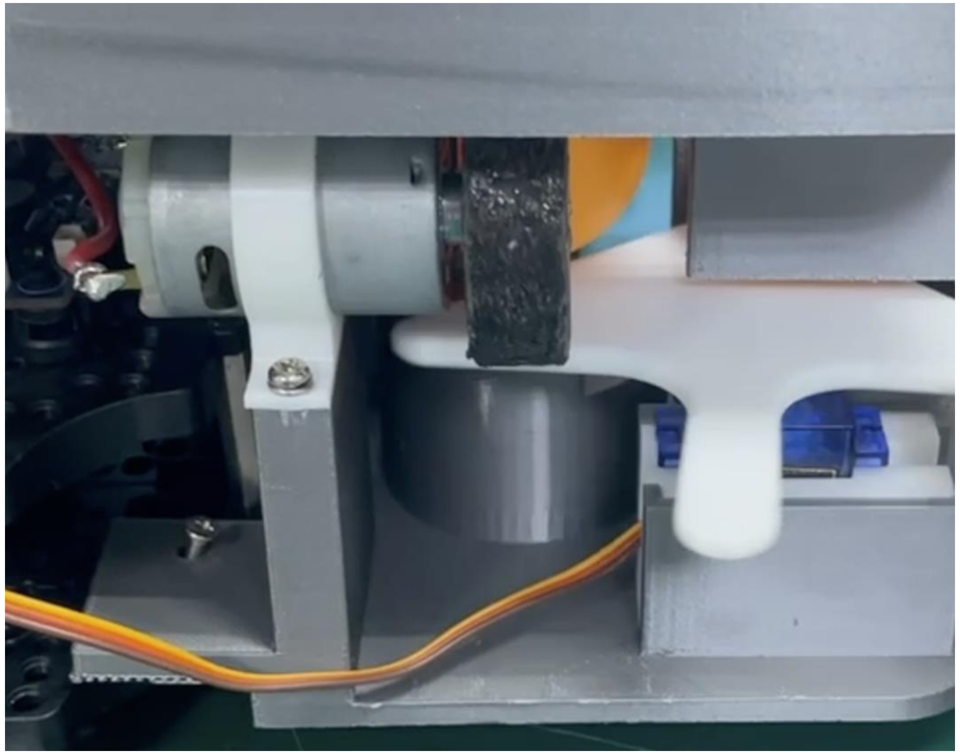
We took a 180 positional servo, attached it to an improvised ball feeder and surprisingly, it worked very well. This ball feeder could feed the ball while blocking the next ball from falling, allowing the feeder to move backwards, eliminating the need for it to move 360

We knew we needed a positional servo for precise control but we could not acquire one that was also 360. We improvised

We took a 180 positional servo, attached it to a make shift ball feeder, propped it up within the larger servo slot (raise it to required height) and surprisingly, it worked very well.

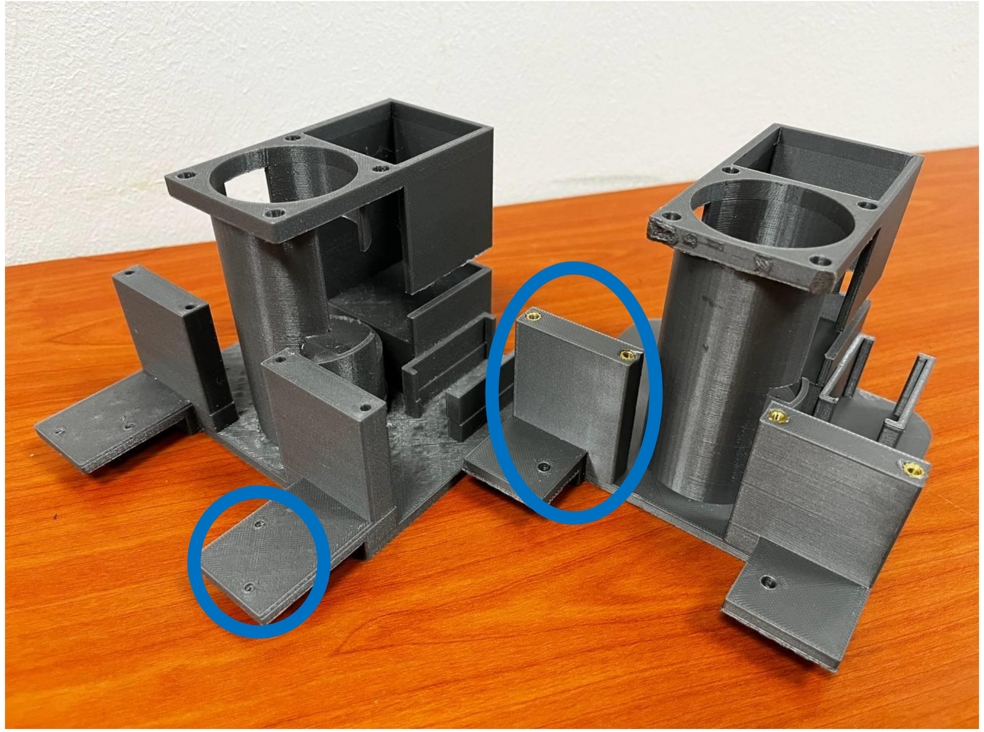
We printed some parts to implement the new ball feeding method

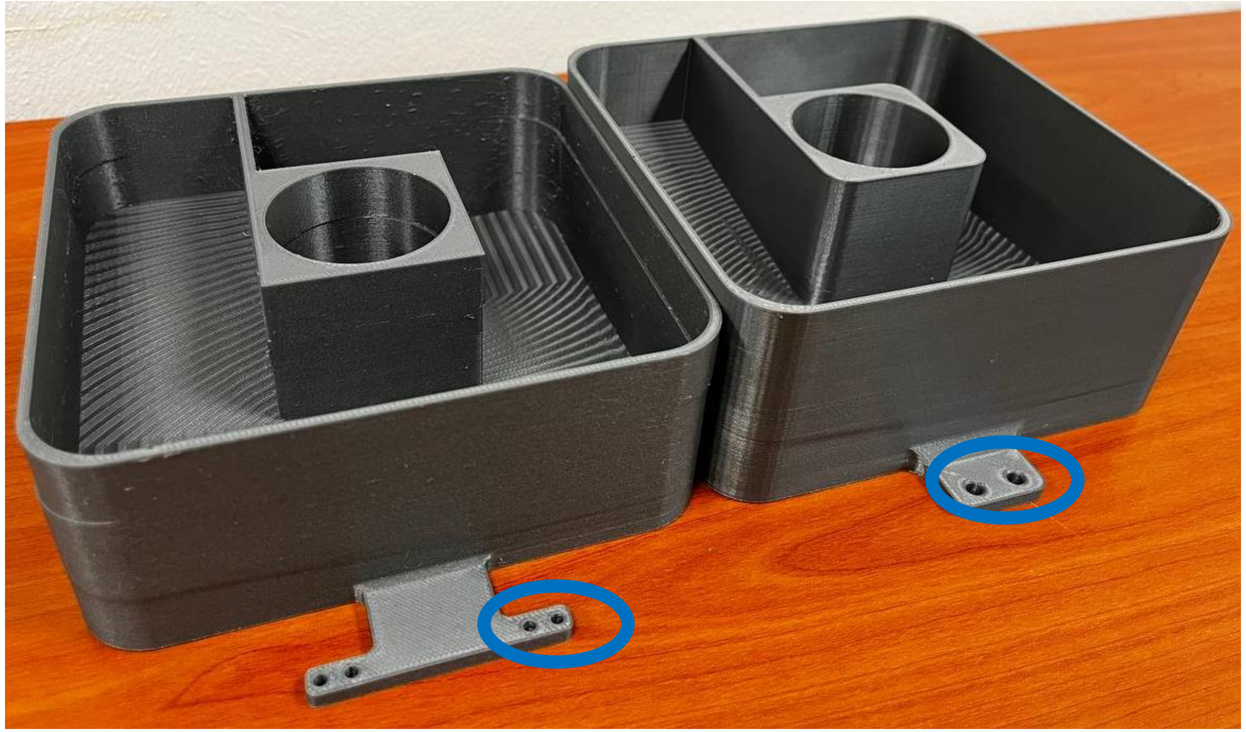




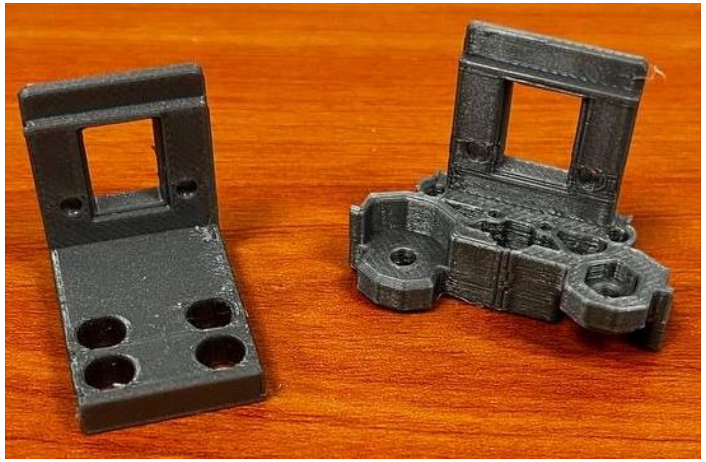
This worked flawlessly

Along the way we made small iterations to improve ease of assembly of the components





- We used M4 screws instead of M2.5 to reduce number of screws required
- We added heat set inserts to improve stability
- We used the waffle plate design for our mounts



So, we arrived at our final design